IN THE CLAIMS:

1. (original) A method of preparing a data packet for transmission over an interconnect link, the method comprising:

calculating a first CRC value for a payload segment of a data packet;

calculating a second CRC value for a sequence number of the data packet;

combining the first CRC value and the second CRC value thereby creating a third CRC value; and

combining the third CRC value with the payload segment of the data packet thereby creating a transmittable data packet.

2. (original) A method as recited in claim 1 further comprising extracting a plurality of inversion bits from the data packet before calculating the first CRC value for the payload segment;

3. (previously presented) A method as recited in claim 1 further comprising extracting the sequence number from the data packet before calculating the second CRC value for the sequence number thereby allowing more space in the payload segment for data.

4. (original) A method as recited in claim 3 wherein the sequence number produces a 30-bit CRC value.

5. (original) A method as recited in claim 1 wherein combining the first CRC value and the second CRC value further includes performing an exclusive OR between the first CRC value and the second CRC value thereby producing the third CRC value.

6. (original) A method as recited in claim 1 wherein combining the third CRC value with the payload segment results in a 77-bit data segment.

7. (previously presented) A method as recited in claim 1 further comprising inserting a plurality of inversion bits into the transmittable data packet, wherein the plurality of inversion bits maintain DC balance and control run length.

8. (currently amended) A method of decoding an encoded message received over an interconnect link in a network, the method comprising:

receiving an encoded data packet having a received CRC segment and a payload segment;

calcuating a first CRC value using the payload segment;

decoding the first CRC value using a plurality of inversion bits;

decoding the received CRC segment using the plurality of inversion bits;

comparing the decoded first CRC value and the decoded received CRC segment thereby determining whether the encoded message was received with any errors.

9. (previously presented) A method as recited in claim 8 further comprising removing the plurality of inversion bits from the encoded message.

10. (previously presented) A method as recited in claim 8 further comprising determining whether an error in the encoded message resulted from a transmission error.

11. (currently amended) A method as recited in claim 10 further comprising comparing a first plurality of preselected bits in the decoded received CRC segment with a second plurality of preselected bits in the decoded first CRC value.

12. (previously presented) A method as recited in claim 11 wherein both the first plurality of preselected bits and the second plurality of preselected bits have the same bit positions.

13. (previously presented) A method as recited in claim 8 further comprising determining whether an error in the encoded message resulted from a non-transmission error.

14. (currently amended) A method as recited in claim 13 further comprising comparing a third plurality of preselected bits in the decoded received CRC segment with a fourth plurality of preselected bits in the decoded first CRC value.

15. (currently amended) A method as recited in claim 13 wherein both the third plurality of preselected bits in the decoded received CRC segment and the fourth plurality of preselected bits in the ~~fourth plurality of preselected bits~~ decoded first CRC value have the same bit positions.

16. (cancelled)

17. (currently amended) A method as recited in claim 8 wherein the decoded first CRC value and the decoded received CRC segment have 30 bits wherein the 30 bits includes a CRC value for a sequence number.